PATENT ABSTRACTS OF JAPAN

(11)Publication number : 06-068075
(43)Date of publication of application : 11.03.1994

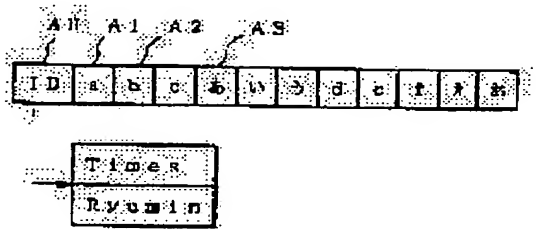
(51)Int.Cl. G06F 15/20
G06F 15/20
G06F 3/153
G09G 5/22

(21)Application number : 04-217827
(22)Date of filing : 17.08.1992

(71)Applicant : TOSHIBA CORP
(72)Inventor : OGIKUBO TOMOFUMI

(54) FONT DATA STORAGE SYSTEM

(57)Abstract:
PURPOSE: To efficiently perform even copy processing and paste processing by efficiently storing data in a short time.
CONSTITUTION: In a system for storing the character data of character strings whose font formats are different in a text buffer, the text buffer is provided with character code areas A1, A2,... for storing the respective character codes of the character strings and a font format area AH for storing all the font format codes of the character strings altogether provided at the head of the areas.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19)日本国特許庁（ＪＰ）　(12)**公開特許公報**（Ａ）　(11)特許出願公開番号

特開平6－68075

(43)公開日　平成６年(1994)３月11日

(51)Int.Cl.[5]		識別記号	庁内整理番号	ＦＩ	技術表示箇所
Ｇ０６Ｆ	15/20	５３０　Ｅ	9288－5L		
		５６２　Ｎ	9288－5L		
	3/153	３１０　Ｂ	7165－5B		
Ｇ０９Ｇ	5/22		9061－5G		

審査請求　未請求　請求項の数１(全 14 頁)

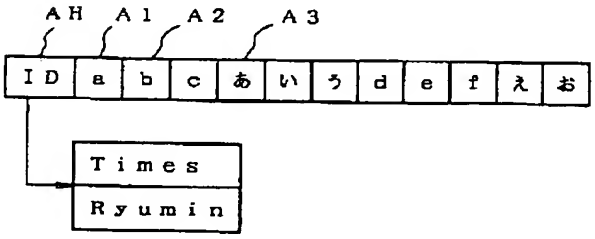
(21)出願番号　特願平4－217827

(22)出願日　平成４年(1992)８月17日

(71)出願人　000003078
株式会社東芝
神奈川県川崎市幸区堀川町72番地
(72)発明者　荻久保　友史
東京都府中市東芝町１番地　株式会社東芝府中工場内
(74)代理人　弁理士　三好　秀和　(外１名)

(54)【発明の名称】　フォントデータ格納方式

(57)【要約】
【目的】　短時間で効率の良いデータ格納処理を可能とし、これによりコピー処理やペースト処理をも効率的に行うことを可能とする。
【構成】　フォント書式の異なる文字列の文字データをテキストバッファに格納する方式において、前記テキストバッファは、前記文字列の各文字コードを格納する文字コードエリアＡ１，Ａ２，…と、このエリアの先頭に設けられ前記文字列の全てのフォント書式コードを一括して格納するフォント書式エリアＡＨとを有する。



1

【特許請求の範囲】

【請求項１】　フォント書式の異なる文字列の文字データをテキストバッファに格納する方式において、
前記テキストバッファは、前記文字列の各文字コードを格納する文字コードエリアと、このエリアの先頭に設けられ前記文字列の全てのフォント書式コードを一括して格納するフォント書式エリアとを有することを特徴とするフォントデータ格納方式。

【発明の詳細な説明】

【０００１】

【産業上の利用分野】本発明は、ＤＴＰ（Desk Top Publishing)等においてフォント書式の異なる文字列を表示する場合に使用されるテキストバッファに対するフォントデータ格納方式に関する。

【０００２】

【従来の技術】ＤＴＰ等においては、フォント書式の異なる多種類のフォントを用い、ひらがな文字や英文字等が混在した文書を作成し、画面表示またはプリントアウトする場合が多い。従来、例えば、図１０に示すように、フォント書式の異なる英数字とひらがな文字から成る文字列“ａｂｃあいうｄｅｆえお”を画面表示またはプリントアウトさせるためには、以下のようにしてテキストバッファにデータを格納していた。

【０００３】今、英数字のフォント名が“Ｔｉｍｅｓ（タイムズ）”であり、ひらがな文字のフォント名が“Ｒｙｕｍｉｎ（リューミン；明朝体）であるとする。この場合、テキストバッファには、図１１に示すように、フォント名の変わり目のエリアにフォント名を示すフォントＩＤを格納しておき、このフォント名の後に、そのフォント名の文字コードを格納する。図１１においては、先頭エリアに英数字“ａｂｃ”のフォント名“Ｔｉｍｅｓ”を示すコードを格納し、その次のエリアから“ａｂｃ”各英数字のコードを格納する。次にひらがな文字“あいう”のフォント名“Ｒｙｕｍｉｎ”を示すコードを格納し、その次のエリアからひらがな文字“あいう”の各ひらがな文字コードを格納する。さらに英数字“ｄｅｆ”のフォント名コード“Ｔｉｍｅｓ”を格納し、その次に各英数字コードを格納する。このように、フォント名が変わる都度、各文字コード列の先頭エリアにフォント名を格納するようしていた。

【０００４】その手順を図１２のフローチャートに基づいて更に詳説する。

【０００５】初めにフラグ１およびフラグ２を“０”にするとともに、テキストバッファをクリアする（ステップＳＴ１０１）。

【０００６】次にテキストバッファの先頭エリアに英数字フォントＩＤを挿入する（ステップＳＴ１０２）。次にオペレータにより文字が入力されるとその文字が英数字か否かが判定される（ステップＳＴ１０３，ＳＴ１０４）。若し、入力文字が英数字であれば、フラグ２を

2

“０”にセット（ステップＳＴ１０５）し、フラグ１＝フラグ２を確認する。最初は、フラグ１＝フラグ２＝“０”が成立する（ステップＳＴ１０６Ｙ）ので、テキストバッファに文字を追加していく（ステップＳＴ１０７）。そして、フラグ２の内容（この場合は“０”）をフラグ１にセットした後、ステップＳＴ１０３に戻り、再び文字入力を実行する（ステップＳＴ１０８）。

【０００７】英数字入力中に英数字以外の文字が入力された場合（ステップＳＴ１０４Ｎ）には、フラグ２を“１”にセット（ステップＳＴ１０９）し、フラグ１＝フラグ２を確認する（ステップＳＴ１０６）。この場合には、フラグ１≠フラグ２であり、かつフラグ２＝“１”である（ステップＳＴ１０６Ｎ，ＳＴ１１０Ｎ）から、テキストバッファに漢字用フォントＩＤを追加する（ステップＳＴ１１１）。その後、テキストバッファに入力された漢字またはひらがな文字を追加し（ステップＳＴ１０７）、フラグ２の内容（＝“１”）をフラグ１にセットした後、再びステップＳＴ１０３に戻る（ステップＳＴ１０８）。

【０００８】さらに次に入力された文字が英数字に変わっている場合（ステップＳＴ１０４Ｙ）には、フラグ２（＝“１”）を“０”にセット（ステップＳＴ１０５）し、フラグ１＝フラグ２を確認する。このときには、フラグ１≠フラグ２であり、かつフラグ２＝“０”である（ステップＳＴ１０６Ｎ，ＳＴ１１０Ｙ）から、テキストバッファに英数字用フォントＩＤを追加する（ステップS.T１１２）。その後、テキストバッファに入力された英数字を追加し（ステップＳＴ１０７）、フラグ２の内容（＝“１”）をフラグ１にセットした後、再びステップＳＴ１０３に戻る（ステップＳＴ１０８）。

【０００９】このように複雑な処理を実行した後、テキストバッファにフォントＩＤが格納されるのである。

【００１０】図１３には、フォントＩＤが“Ｒｙｕｍｉｎ”と“Ｔｉｍｅｓ”から成る文字列「ａｂｃあいうｄｅｆえおｇｈｉかきくｊｋｌけこｍｎｏさしすｏｐｒせそ」を格納するファイル（ＭＩＦファイル）形式の従来例を示す。図から理解されるように、フォントＩＤが変更される都度、フォントＩＤを付加するので、その手順は複雑なものとなっている。

【００１１】次に、表示された文字列の中でマウス等で選択された部分のフォント書式をコピーして異なるフォント書式（例えば、Ｈｅｌｖｅｔｉｃａ，Ｇｏｔｈｉｃ）から成る他の文字列へペーストする処理の従来例について説明する。

【００１２】例えば、図１０において、範囲Ａの部分「ｅｆえ」をコピーする場合には、英字の“ｅｆ”は“Ｔｉｍｅｓ”であり、ひらがなの“え”は“Ｒｙｕｍｉｎ”であり、英数字フォントＴｉｍｅｓと日本語フォントＲｙｕｍｉｎをそれぞれ取り出さなければならない。図１４は従来のフォントコピー処理の手順を示し、

3

図１５は従来のフォント書式ペースト処理の手順を示している。

【００１３】図１４において、先ず文字列の中からコピーしたい部分を選択してテキストバッファ内へ取り込む（ステップＳＴ１２１）。図１０の例では、「ｅｆえ」が取り込まれる。次に、バッファの中からフォントＩＤが検索、抽出される（ステップＳＴ１２２）。図１０の例では、“Ｔｉｍｅｓ，Ｒｙｕｍｉｎ”が検索され抽出される。この場合、英字の“ｅｆ”は“Ｔｉｍｅｓ”であり、ひらがなの“え”は“Ｒｙｕｍｉｎ”であり、英字フォントＩＤ“Ｔｉｍｅｓ”と日本語フォントＩＤ“Ｒｙｕｍｉｎ”をそれぞれ取り出さなければならない。こうして、英数字フォントＩＤおよび漢字フォントＩＤが全て揃うと（ステップＳＴ１２３Ｙ）、抽出したフォントＩＤをテキストバッファのクリップボードに格納する（ステップＳＴ１２４）。若し、選択部分のＩＤが見付からない場合には、テキストバッファを遡ってフォントＩＤを抽出する（ステップＳＴ１２５）。

【００１４】このようにしてコピーされたフォント情報を他の文字列にペーストする場合は、図１５において、先ず、ペースト先の文字列の中からマウス等で選択された部分をバッファに取り込む（ステップＳＴ１３１）。次に、この選択部分の英数字フォントＩＤおよび漢字フォントＩＤを全て検索し抽出する（ステップＳＴ１３２）。次いで、バッファの先頭が英数字の場合（ステップＳＴ１３３Ｙ）には、英数字フォントＩＤをバッファの先頭に挿入する（ステップＳＴ１３４）。バッファの先頭が英数字でない場合（ステップＳＴ１３３Ｎ）には、漢字フォントＩＤをバッファの先頭に挿入する（ステップＳＴ１３５）。

【００１５】次に、バッファを走査しフォントＩＤを抽出する（ステップＳＴ１３６）。抽出されたフォントＩＤが英数字フォントＩＤであれば（ステップＳＴ１３７Ｙ）、先にクリップボード内に保存されている英数字フォントＩＤと交換する（ステップＳＴ１３８）。抽出されたフォントＩＤが英数字フォントＩＤでなければ（ステップＳＴ１３７Ｙ）、先にクリップボード内に保存されている漢字フォントＩＤと交換する（ステップＳＴ１３９）。こうしてバッファの走査が終了すると（ステップＳＴ１４０Ｙ）、ステップＳＴ１３２の処理で抽出されたフォントＩＤをバッファの最後尾に追加する（ステップＳＴ１４１）。最後にこのようにして編集された選択部分を当初のバッファの対応部分と入れ替える（ステップＳＴ１４２）。これにより、選択部分が新たなフォントＩＤを含むデータとなり、ペースト処理が終了する。

【００１６】

【発明が解決しようとする課題】このように、従来のコード格納方式によれば、図１２のフローチャートおよび図１３に示したような複雑の入力処理手順を踏んでおり、頻繁に英字・漢字が交互に現れてくるような文字列の場合には、多くのフォント情報が必要となり、データ格納処理に長い時間がかかり処理効率が悪いという問題がある。また、図１４および図１５のフローチャートに示したようなコピー処理およびペースト処理も複雑となり、効率的でない。

4

【００１７】本発明は上記の事情に鑑みてなされたものであり、その目的は、短時間で効率の良いデータ格納処理を可能とし、これによりコピー処理やペースト処理をも効率的に行うことができるフォントデータの格納方式を提供することにある。

【００１８】

【課題を解決するための手段】上記の目的を達成するために本発明は、フォント書式の異なる文字列の文字データをテキストバッファに格納する方式において、前記テキストバッファは、前記文字列の各文字コードを格納する文字コードエリアと、このエリアの先頭に設けられ前記文字列の全てのフォント書式コードを一括して格納するフォント書式エリアとを有することを特徴とする。

【００１９】

【作用】上記構成によれば、フォント書式コード（フォントＩＤ）が全てテキストバッファの先頭エリアに格納されているので、フォント書式が異なる毎にフォントＩＤを挿入している従来に比較して、格納処理手順が簡単になるとともに、コピー処理やペースト処理を効率良く行うことができる。

【００２０】

【実施例】図１は本発明方式が適用されたテキストバッファの構成を示し、図１は本発明方式が適用されたパーソナルコンピュータの全体構成を示している。

【００２１】初めに、このパーソナルコンピュータの全体構成を説明すると、キーボードやマウス等から成る入力部１と、入力されたデータを編集する編集処理部２と、編集処理部２で編集されたデータ等を保存する保存部３と、表示用データを作成する表示処理部４と、ＣＲＴを含み、作成された表示用データを画面表示するための画面処理部５と、プリンタを含み、作成された表示用データから印刷用データを生成してプリントアウトする印刷処理部６とから構成されている。前記編集処理部２は、さらに、文書編集部２１と図形処理部２２とから成り、また表示処理部４はテキスト表示処理部４１と図形表示処理部４２とから成っている。

【００２２】図１に示すテキストバッファは、前記文書編集部２１内に設けられ、先頭エリア（フォント書式エリア）ＡＨにフォント情報のポインタＩＤ（フォントＩＤ）を一括格納し、その後に続くエリアＡ１，Ａ２，Ａ３，…に表示文字のコードを格納している。すなわち、先頭エリアＡＨには、その後に続く表示文字列で使用されるフォント情報のＩＤが全て格納されている。図の例では、ＴｉｍｅｓとＲｙｕｍｉｎを格納している。

5

【００２３】図３は、このテキストバッファにフォントＩＤを格納するための入力処理の手順を示している。

【００２４】先ずテキストバッファがクリアされ（ステップＳＴ１）た後、テキストバッファの先頭エリアＡＨにフォントＩＤが挿入される（ステップＳＴ２）。このＩＤは、表示される文字列の全てのフォントＩＤを含んでいる。

【００２５】次にテキストバッファのエリアＡ１，Ａ２，Ａ３，…に順番に文字列のコードを入力していけば良い（ステップＳＴ３～ＳＴ４）。

【００２６】図４は、前記従来例の図１３に対応するＭＩＦファイル形式を示しており、両図を比較すると良く分かるように、本実施例ではその処理手順が極めて簡素化され、そのデータ量が大幅に縮小される。

【００２７】次にこのように格納されたテキストバッファの情報からその文字列を表示する処理を図５のフローチャートに基づいて説明する。この処理は表示処理部４のテキスト表示処理部４１で実行される。

【００２８】初めに、フラグ１およびフラグ２をともに“０”クリアし（ステップＳＴ１１）、仮に、英数字フォントを設定する（ステップＳＴ１２）。次にテキストバッファから１文字を読み込み、これが英数字であれば（ステップＳＴ１４Ｙ）、フラグ２＝“０”を確認し（ステップＳＴ１５）、次いで、フラグ１＝フラグ２を条件として文字を表示する（ステップＳＴ１７）。その後、ステップＳＴ３に戻り、次に１文字をテキストバッファから読み込む。このとき、読み込まれた文字が英数字でなければ（ステップＳＴ１４Ｎ）、フラグ２を“１”にセットする（ステップＳＴ１９）。次に、フラグ１＝フラグ２を確認するが、このときには、フラグ１≠フラグ２であり、かつフラグ２≠“１”であるので、漢字フォントが設定される（ステップＳＴ１６Ｎ，ＳＴ２０Ｎ，ＳＴ２１）。そして、フラグ１の内容をフラグ２にセットした後（ステップＳＴ２２）、文字表示がされる（ステップＳＴ１７）。その後、ステップＳＴ３に戻り、次に１文字をテキストバッファから読み込む。このとき、読み込まれた文字が英数字であれば、フラグ２を“０”にする（ステップＳＴ１５）。次に、フラグ１＝フラグ２であるか否かを判定するが、このとき、前回表示処理でフラグ１が“１”となっているので、ステップＳＴ２０に進む。ここでフラグ２＝０が判別されるが、このときにはフラグ２＝０となっているので、英数字フォントが設定される（ステップＳＴ２３）。そして、フラグ１の内容をフラグ２にセットした後（ステップＳＴ２２）、文字表示がされる（ステップＳＴ１７）。その後、ステップＳＴ３に戻り、次に１文字をテキストバッファから読み込む。このようにして、フォント情報に対応した文字列が表示されるのである。

【００２９】次に、このようにして表示された文字列からフォントをコピーして他の文字列へペーストする処理

6

について説明する。

【００３０】例えば、図６に示すように、文字列「ａｂｃｄｅｆｇあいうえおかき」のフォントが“Ｒｙｕｍｉｎ”と“Ｔｉｍｅｓ”で構成され、文字列「ｈｉｊｋｌｍｎさしすせそたち」のフォントが“Ｇｏｔｈｉｃ”と“Ｈｅｌｖｅｔｉｃａ”で表示されている場合に、図中斜線で囲む選択部分（表示画面中では反転部分）「ｆｇあい」のフォント書式をコピーして、「ｍｎさしす」へペーストする処理を例に説明する。

【００３１】図７において、先ず、選択部分「ｆｇあい」をテキストバッファに取り込み、テキストバッファの中からのフォントＩＤを検索して抽出する（ステップＳＴ３１，ＳＴ３２，ＳＴ３３）。このとき抽出されたフォントＩＤは“Ｔｉｍｅｓ，Ｒｙｕｍｉｎ”である。次に、抽出されたフォントＩＤをクリップボードに格納する（ステップＳＴ３４）。なお、若しフォントＩＤがみつからない場合には、選択部分を遡ってフォントＩＤを抽出する（ステップＳＴ３５）。

【００３２】次に図８のペースト処理を実行すると、ペースト先の選択部分「ｍｎさしす」をバッファに取り込み、そのテキストバッファのフォントＩＤを検索、抽出する（ステップＳＴ４１，４２）。このとき抽出されたフォントＩＤは“Ｇｏｔｈｉｃ，Ｈｅｌｖｅｔｉｃａ”である。次に前記クリップボードのフォントＩＤ（Ｒｙｕｍｉｎ；Ｔｉｍｅｓ））をテキストバッファの先頭に挿入するとともに、抽出されたフォントＩＤ（Ｇｏｔｈｉｃ，Ｈｅｌｖｅｔｉｃａ）をテキストバッファの最後尾に追加する（ステップＳＴ４３，ＳＴ４４）。そして、前記選択部分を先に格納されているテキストバッファの部分と入れ替える（ステップＳＴ４５）。そのときのテキストバッファ内の構成を図９に示す。このように、フォント書式の複写処理を極めて容易に行うことができる。

【００３３】次に本発明のカタログ機能について説明する。

【００３４】このカタログ機能は、あるフォント書式に対して名前を付けておき、テキストに対してその名前を適用することで、簡単にフォントの変更ができるようにしたものである。

【００３５】例えば、フォント書式がＧｏｔｈｉｃ　１０ｐｔ（１０ポイントのゴシック文字）と、Ｈｅｌｖｅｔｉｃａ１２　ｐｔ（１２ポイントのヘルベチカ文字）というフォント書式について、例えば“Ｂｏｄｙ”という名前で予め登録しておき、Ｒｙｕｍｉｎ　１０ｐｔ（１０ポイントの明朝文字）ＴとＴｉｍｅｓ　１２ｐｔ（１２ポイントのタイムズ文字）から成る文字列の選択部分にフォント書式Ｇｏｔｈｉｃ　１０ｐｔ、Ｈｅｌｖｅｔｉｃａ１２　ｐｔをペーストするには、フォント書式“Ｂｏｄｙ”を指定して、文字列の選択部分にペーストすれば良い。本発明の格納方式であれば、このような

7

カタログ機能を極めて容易に行うことができる。

【００３６】

【発明の効果】以上説明したように本発明によれば、短時間で効率の良いデータ格納処理を可能とし、これによりコピー処理やペースト処理をも効率的に行うことができる。

【図面の簡単な説明】

【図１】本発明が適用されたテキストバッファの構成を示す説明図である。

【図２】本発明が適用されたパーソナルコンピュータの全体構成を示す説明図である。

【図３】テキストバッファへの入力格納処理の手順を示すフローチャートである。

【図４】本発明が適用されたＭＩＦファイル形式を示す説明図である。

【図５】本発明における画面表示処理の手順を示すフローチャートである。

【図６】フォント書式のコピー処理およびペースト処理を示す説明図である。

【図７】本発明におけるフォント書式のコピー処理の手順を示すフローチャートである。

【図８】本発明におけるフォント書式のペースト処理の手順を示すフローチャートである。

8

【図９】本発明におけるフォント書式のペースト処理実行後のテキストバッファの構成を示す説明図である。

【図１０】画面上に表示されたフォント書式の異なる文字列を示す説明図である。

【図１１】従来のテキストバッファの構成を示す説明図である。

【図１２】従来におけるフォント書式入力処理の手順を示すフローチャートである。

【図１３】従来におけるＭＩＦファイル形式を示す説明図である。

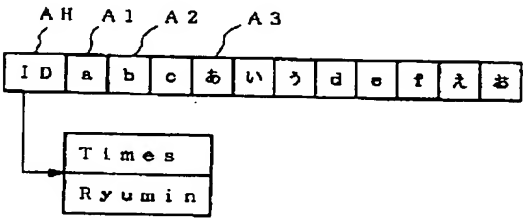
【図１４】従来におけるフォント書式コピー処理の手順を示すフローチャートである。

【図１５】従来におけるフォント書式ペースト処理の手順を示すフローチャートである。

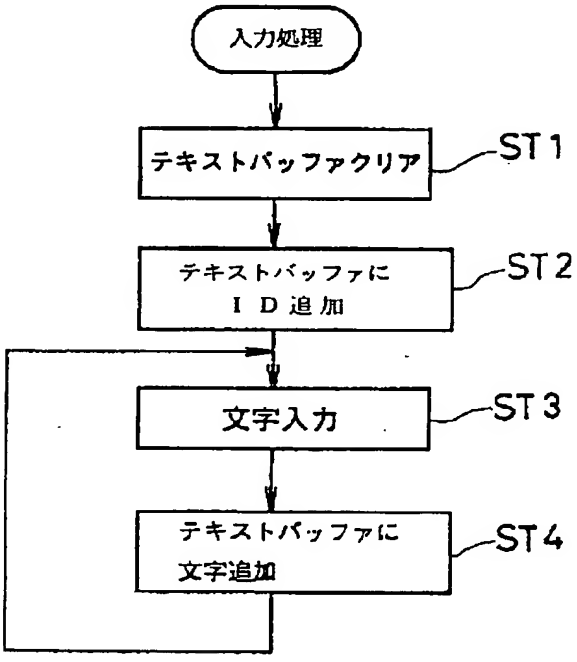
【符号の説明】

1　入力部
2　編集処理部
3　文書保存部
4　表示処理部
5　画面表示処理部
6　印刷処理部
AH　先頭エリア（フォント書式エリア）

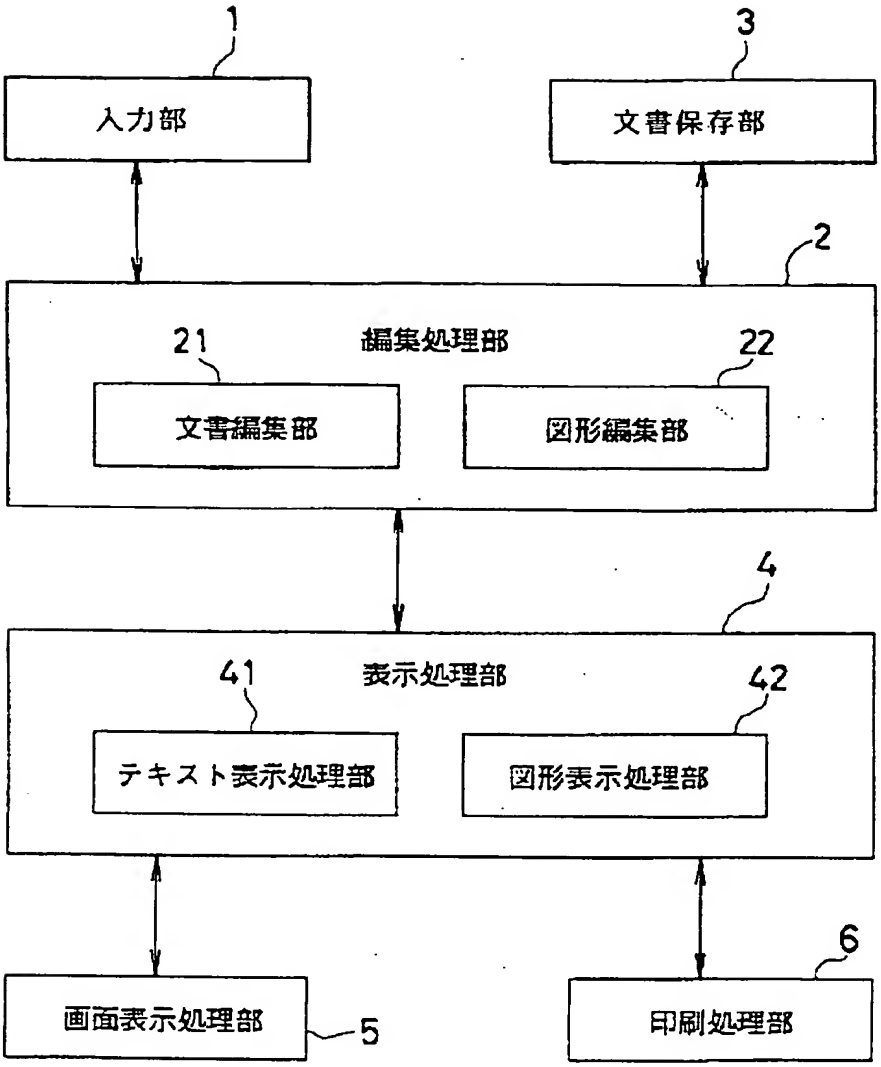
【図１】



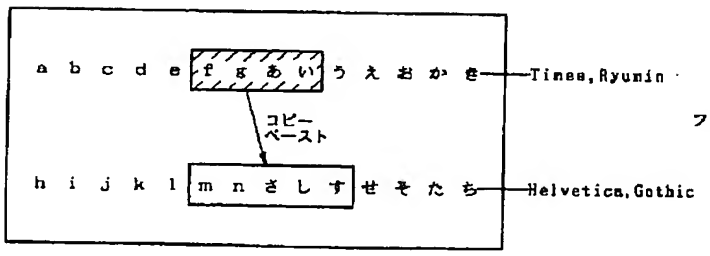
【図３】



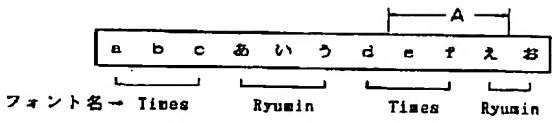
【図２】


1
入力部
3
文書保存部
2
21
編集処理部
22
文書編集部
図形編集部
4
41
表示処理部
42
テキスト表示処理部
図形表示処理部
6
画面表示処理部
5
印刷処理部

【図６】


a b c d e f g あ い う え お か き
Times,Ryumin
コピー
ペースト
h i j k l m n さ し す せ そ た ち
Helvetica,Gothic

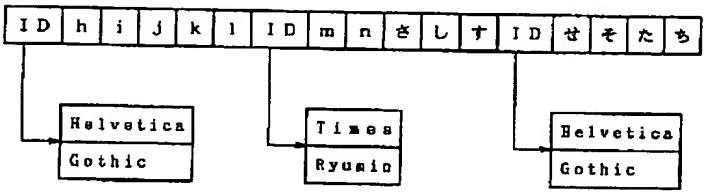
【図１０】


A
a b c あ い う d e f え お
フォント名→ Times
Ryumin
Times
Ryumin

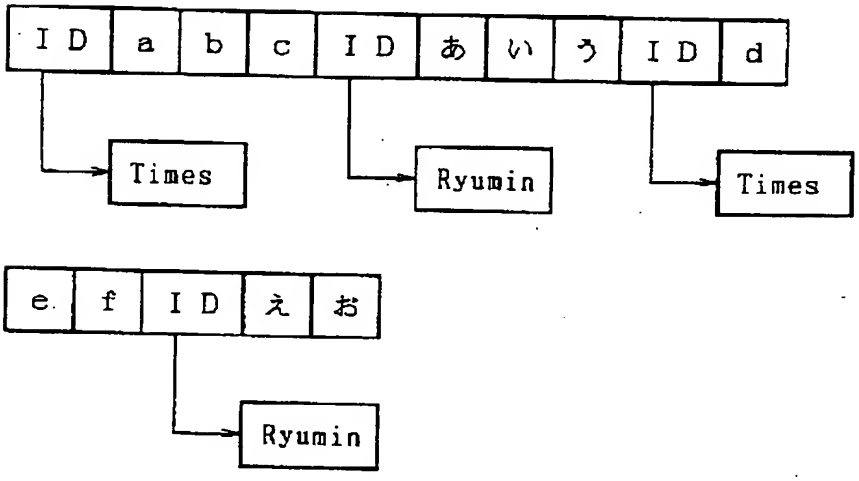
【図４】

```
 <Para
<ParaLine
<String 'abcあいうdefえおghiかきくjklけこmnoさしすoprせそ'>
 >
 > # end of Para
```

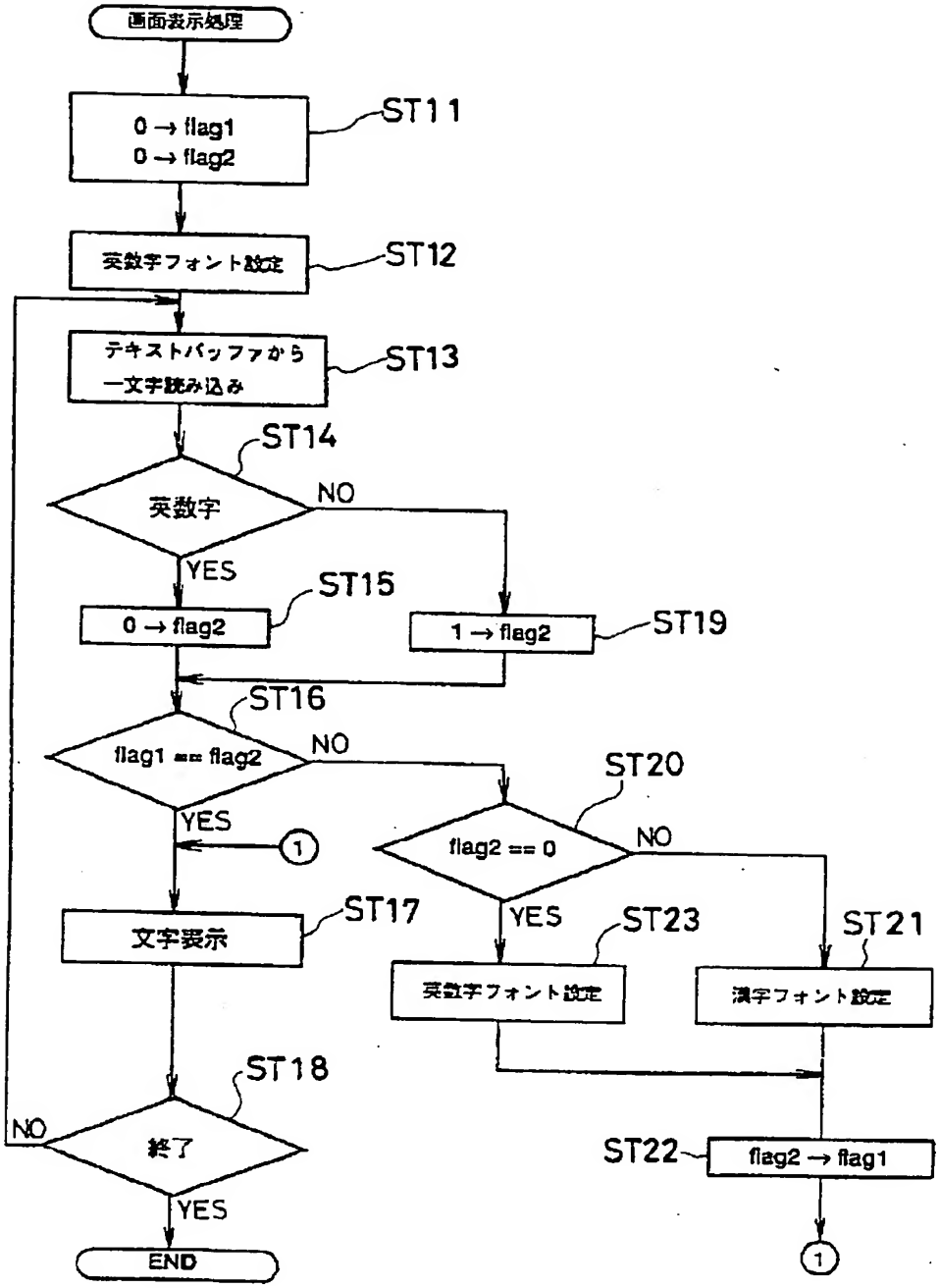
【図９】



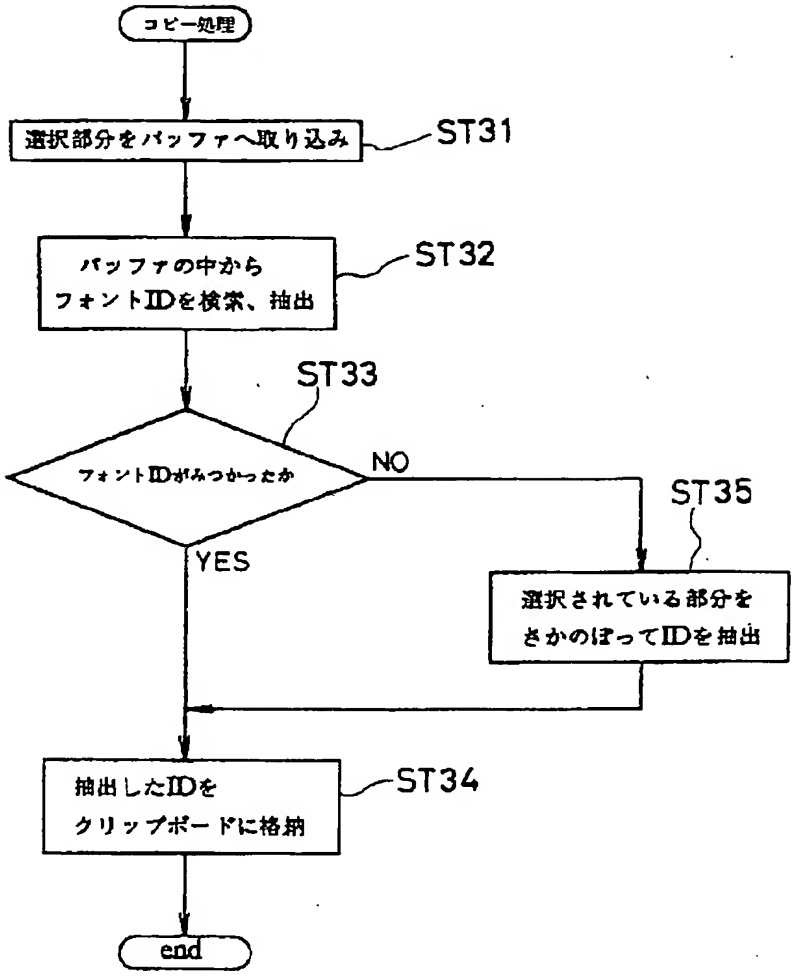
【図１１】



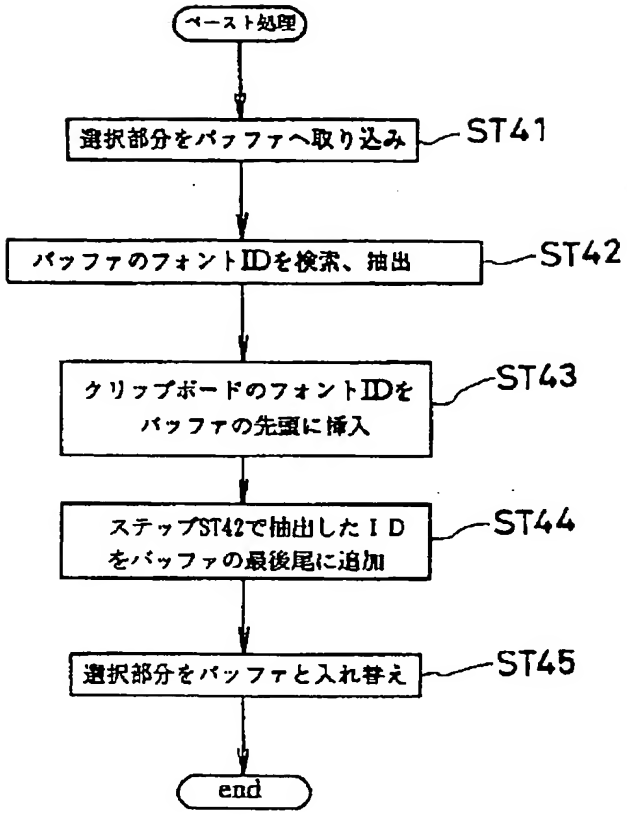
【図5】


画面表示処理
0 → flag1
0 → flag2
ST11
英数字フォント設定
ST12
テキストバッファから
一文字読み込み
ST13
ST14
英数字
NO
YES
ST15
0 → flag2
1 → flag2
ST19
ST16
flag1 == flag2
NO
YES
ST20
flag2 == 0
NO
YES
ST23
ST21
英数字フォント設定
漢字フォント設定
ST17
文字表示
ST18
NO
終了
YES
END
ST22
flag2 → flag1
1
1

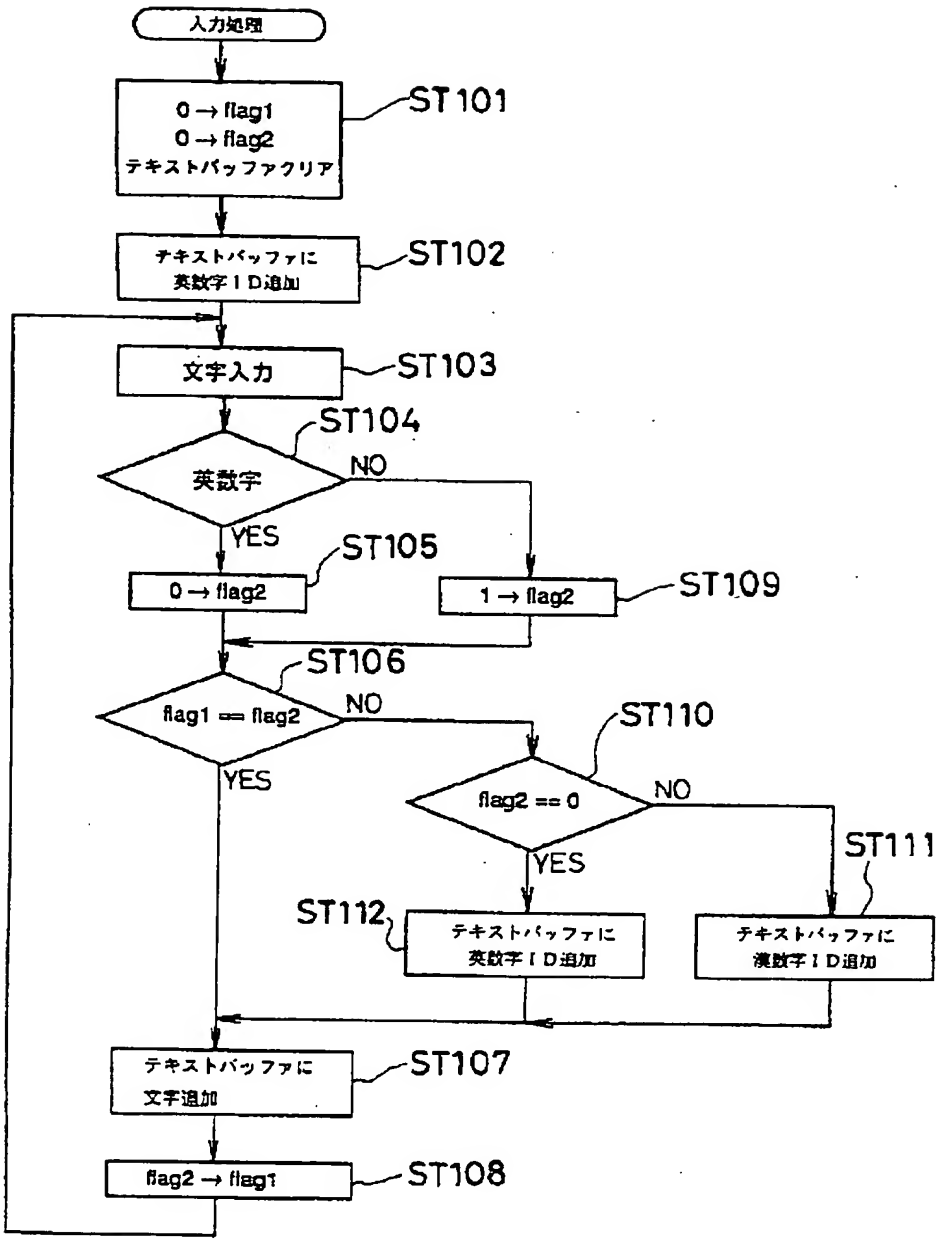
【図７】


コピー処理
選択部分をバッファへ取り込み
ST31
バッファの中から
フォントIDを検索、抽出
ST32
ST33
フォントIDがみつかったか
NO
YES
ST35
選択されている部分を
さかのぼってIDを抽出
抽出したIDを
クリップボードに格納
ST34
end

【図８】


ペースト処理
選択部分をバッファへ取り込み
ST41
バッファのフォントIDを検索、抽出
ST42
クリップボードのフォントIDを
バッファの先頭に挿入
ST43
ステップST42で抽出したＩＤ
をバッファの最後尾に追加
ST44
選択部分をバッファと入れ替え
ST45
end

【図１２】

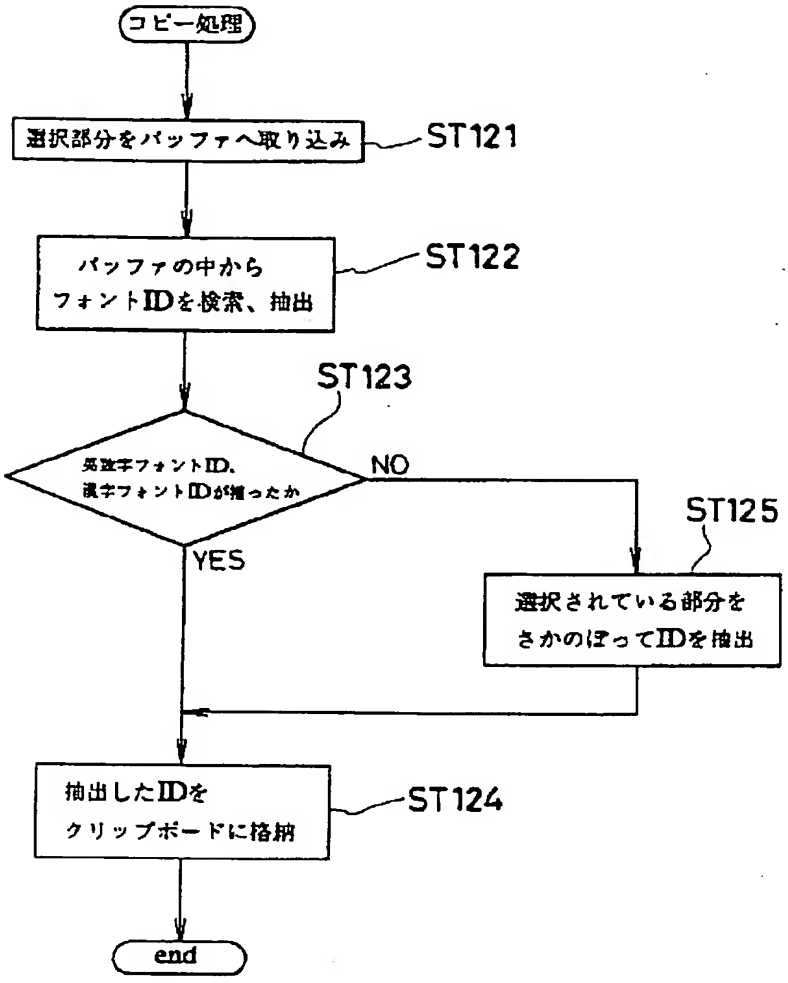

入力処理
0 → flag1
0 → flag2
テキストバッファクリア
ST101
テキストバッファに
英数字ＩＤ追加
ST102
文字入力
ST103
ST104
英数字
NO
YES
ST105
0 → flag2
1 → flag2
ST109
ST106
flag1 == flag2
NO
YES
ST110
flag2 == 0
NO
YES
ST111
ST112
テキストバッファに
英数字ＩＤ追加
テキストバッファに
漢数字ＩＤ追加
テキストバッファに
文字追加
ST107
flag2 → flag1
ST108

【図１３】

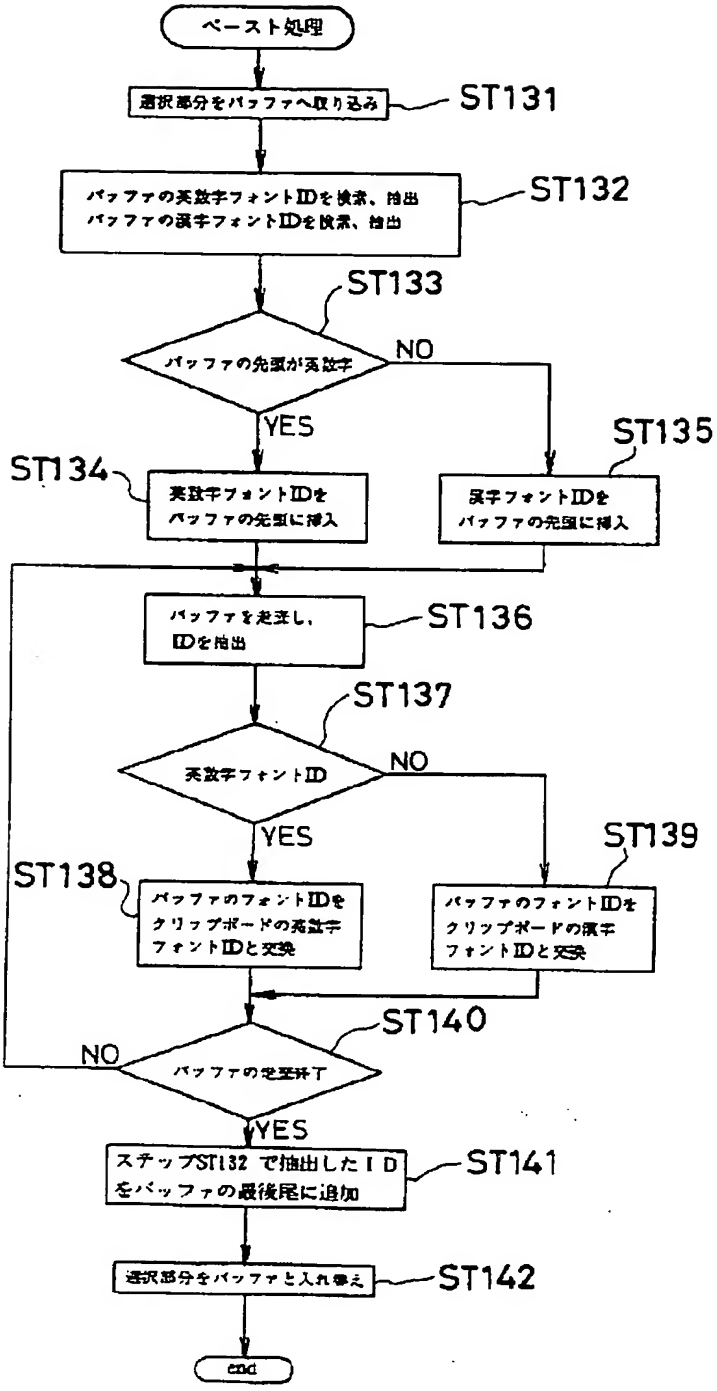
```
 <ParaLine
<String 'abc'>
 <Font
 <FFamily '明朝'>
 <FSize 10>
 > # end of Font
 <String 'あいう'>
 <Font
 <FFamily 'Times'>
 <FSize 12>
 > # end of Font
 <String 'def'>
 <Font
 <FFamily '明朝'>
 <FSize 10>
 > # end of Font
 <String 'えお'>
 <Font
 <FFamily 'Times'>
 <FSize 12>
 > # end of Font
 <String 'ghi'>
 <Font
 <FFamily '明朝'>
 <FSize 10>
 > # end of Font
 <String 'かきく'>
 <Font
 <FFamily 'Times'>
 <FSize 12>
 > # end of Font
 <String 'jkl'>
```

```
 <Font
 <FFamily '明朝'>
 <FSize 10>
 > # end of Font
 <String 'けこ'>
 <Font
 <FFamily 'Times'>
 <FSize 12>
 > # end of Font
 <String 'mno'>
 <Font
 <FFamily '明朝'>
 <FSize 10>
 > # end of Font
 <String 'さしす'>
 <Font
 <FFamily 'Times'>
 <FSize 12>
 > # end of Font
 <String 'opr'>
 <Font
 <FFamily '明朝'>
 <FSize 10>
 > # end of Font
 <String 'せそ'>
 <Font
 <FFamily 'Times'>
 <FSize 12>
 > # end of Font
 >
 > # end of Para
```

【図１４】


コピー処理
選択部分をバッファへ取り込み
ST121
バッファの中から
フォントIDを検索、抽出
ST122
ST123
英数字フォントID、
漢字フォントIDが揃ったか
NO
YES
ST125
選択されている部分を
さかのぼってIDを抽出
抽出したIDを
クリップボードに格納
ST124
end

【図１５】


ペースト処理
選択部分をバッファへ取り込み
ST131
バッファの英数字フォントIDを検索、抽出
バッファの漢字フォントIDを検索、抽出
ST132
ST133
バッファの先頭が英数字
NO
YES
ST134
英数字フォントIDを
バッファの先頭に挿入
ST135
漢字フォントIDを
バッファの先頭に挿入
バッファを走査し、
IDを抽出
ST136
ST137
英数字フォントID
NO
YES
ST138
バッファのフォントIDを
クリップボードの英数字
フォントIDと交換
ST139
バッファのフォントIDを
クリップボードの漢字
フォントIDと交換
ST140
NO
バッファの走査終了
YES
ステップST132で抽出したID
をバッファの最後尾に追加
ST141
選択部分をバッファと入れ換え
ST142
end